## Amendments to the Claims

This Listing of Claims will replace all prior versions, and listings, of claims in this application:

**Listing of Claims:**

1. (Currently Amended) A computer-implemented method for processing a p-code file, comprising:

analyzing p-code methods ~~to be compiled~~ within said p-code file to determine a resource utilization for the p-code methods;

identifying one or more p-code methods that have ~~at least one profile~~a resource utilization parameter above a threshold level; and

annotating said identified p-code methods to be compiled, said annotating comprising inserting an in-line ~~associating a respective~~ priority level hint ~~with~~ for each annotated p-code method ~~to be compiled~~, said priority level hints being hierarchically-related and collectively representing a hierarchical order, said priority level hints enabling preferential processing of said p-code methods in a hierarchical manner corresponding to said hierarchical order of said priority level hints.

2. (Canceled)

3. (Previously Presented) The method of claim 1, wherein:

said p-code file comprises an application for processing by a virtual machine (VM) just-in-time (JIT) compiler.

4. (Canceled)

5. (Currently Amended) The method of claim 1, ~~wherein~~ further comprising providing said priority level hints ~~are provided~~ as a separate file.

6. (Currently Amended) The method of claim 1, wherein:

said ~~at least one profile~~ resource utilization parameter comprises at least one of a method execution time, a frequency of method invocation, a number of instructions and a use of loop structures.

7. (Currently Amended) The method of claim 1, wherein:

said ~~at least one profile~~ resource utilization parameter comprises at least one of an execution time parameter, an input/output utilization parameter and a processor utilization parameter.

8-9. (Canceled)

10. (Previously Presented) The method of claim 1, wherein:

said annotating comprises selectively setting each of a plurality of normally unused bits within a method access flag field of an identified class file, wherein said unused bits are selectively set to define thereby said priority level hint of a respective annotated method.

11. (Currently Amended) The method of claim ~~3~~1, wherein:

each identified ~~byte-code portion of said application~~p-code method is associated with one of a plurality of priority levels, said priority level hints being indicative of respective priority levels.

12. (Currently Amended) The method of claim 3, further comprising: selectively pre-compiling ~~at least a portion~~portions ~~of said application~~ p-code file that are designated by the in-line hints.

13. (Previously Presented) The method of claim 12, wherein: said precompiled portion of said application file is included within a virtual machine.

14-32. (Canceled)

33. (New) The method of claim 1, further comprising replacing lines of interpreted instructions in the p-code file with compiled code for the identified p-code methods.

34. (New) The method of claim 1, further comprising managing storage of methods in a cache memory according to the hierarchical order.

35. (New) A computer-readable medium storing computer-executable instructions for performing the following method for processing a p-code file, comprising:

analyzing p-code methods within said p-code file to determine a resource utilization for the p-code methods;

identifying one or more p-code methods that have a resource utilization parameter above a threshold level; and

annotating said identified p-code methods to be compiled, said annotating comprising inserting an in-line priority level hint for each annotated p-code method , said priority level hints being hierarchically-related and collectively representing a hierarchical order, said priority level hints enabling preferential processing of said p-code methods in a hierarchical manner corresponding to said hierarchical order of said priority level hints.

36. (New) The computer-readable medium of claim 35, further storing computer-exectuable instructions to provide said priority level hints as a separate file.

37. (New) The computer-readable medium of claim 35, wherein:

said resource utilization parameter comprises at least one of a method execution time, a frequency of method invocation, a number of instructions and a use of loop structures.

38. (New) The computer-readable medium of claim 35, wherein:

said resource utilization parameter comprises at least one of an execution time parameter, an input/output utilization parameter and a processor utilization parameter.

39. (New) The computer-readable medium of claim 35, wherein:

said annotating comprises selectively setting each of a plurality of normally unused bits within a method access flag field of an identified class file, wherein said unused bits are selectively set to define thereby said priority level hint of a respective annotated method.

40. (New) The computer-readable medium of claim 35, wherein:

each identified p-code method is associated with one of a plurality of priority levels, said priority level hints being indicative of respective priority levels.

41. (New) The computer-readable medium of claim 35, further storing computer-executable instructions for performing the following:

selectively pre-compiling portions of said p-code file that are designated by the in-line hints.

42. (New) The computer-readable medium of claim 35, further storing computer-executable instructions for performing the following:

replacing lines of interpreted instructions in the p-code file with compiled code for the identified p-code methods.

43. (New) The computer-readable medium of claim 35, further storing computer-executable instructions for performing the following:

managing storage of methods in a cache memory according to the hierarchical order.